the same and has 3 pins "1", "2" and "3". The first pin "1" in the passcode appears in the last token in the 4×4 table [G5 716], and that token should be selected as the first key token. However, the second token [G5 718] in the table was selected, and is shown in the first key token position G5 721. This is wrong and would invalidate this message.

In the screenshot, when the user clicks the "Check" button G5 726, it shows that the validation process has failed [G5 731] for this character "P".

FIG. 20B is a sample screenshot of a message decryption process using the GATE_5 embodiment. It shows what happens behind the scenes when the process shown in FIG. 20A is taking place and how a filler message is invalidated on the receiver side. The process proceeds as follows:

Message character "P" [G5 751] is encrypted with sender passcode "123" [G5 752] and attached with a 4×4 table of 16 tokens [G5 755] and a key with some tokens [G5 721, G5 723 and G5 725]. This information is sent to the network [G5 757] then to the receiver [G5 759].

On the receiver side, it uses the same passcode "123" [G5 760] to decrypt the message. The key tokens go through [G5 763] the validation process G5 765, G5 767, to check each key token. From K61, K62 and K63 one can see the 1st token is invalid.

The 1st passcode "1" appeared in the last token [G5 790], and it should be selected. However, the 2nd token [G5 792] was selected and shows up in the first key token position [G5 721]. It is therefore invalid.

A final conclusion is reached [G5 769] that the message is invalid [G5 771], as shown in FIG. 20A [G5 731].

FIG. 21 is a sample screenshot of a message decryption process using the GATE_5 embodiment. It shows an example of how an encrypted original message may not be successfully decrypted by a receiver passcode that is different from the sender passcode. The process proceeds as follows:

User typed in a plain text message "FYEO" in G5 700, then entered passcode "123" [G5 702] and clicked on the "Encrypt" button [G5 703].

The message is encrypted, sent and received, and showed up at G5 705 as: "FlPRojcYnEbAO" [G5 705].

Receiver uses passcode "680" [G5 707] to decrypt the message received from the sender, which is encrypted with passcode "123" [G5 702].

The decrypted message is highlighted in G5 705: "nE".

The result message is shown as "nE" [G5 709].

The result message is different from the original message from sender: "FYEO" [G5 700], because the receiver used a different passcode to decrypt the message.

The following steps are preferably used, for each pin in the passcode, to generate valid key tokens against a 4×4 table with 16 tokens for each valid message:

Go through all the 16 tokens: (a) if the pin is found in a token, pick that token; and (b) if the pin is not in any token, pick a random token from the 16 tokens in the table.

The following steps are preferably used to generate invalid key tokens against a 4×4 table with 16 tokens for each invalid message:

<1> Set boolean "Done_Fixing" to false

<2> Go through all the 16 tokens, do steps <3> and <4> below for each pin in the passcode

<3> <A> If the pin is found in a token:

(1) If Done_Fixing equals false, pick any other token except this one to intentionally pick a wrong token, and set Done_Fixing to true.

(2) If Done_Fixing equals true, pick that token.

<B> If the pin is not in any token, pick a random token from the 16.

<4> Save the key token generated above into a vector.

<5> Generate a random number N in the range of: −1 to 1

<A> If N=−1, delete the last key token from the vector.

<B> If N=0, do nothing.

<C> If N=1 and user pin length <6, add a random token from the 16 in the table to the vector.

<6> The tokens in the vector will be the final key tokens.

The foregoing embodiments and advantages are merely exemplary, and are not to be construed as limiting the present invention. The description of the present invention is intended to be illustrative, and not to limit the scope of the claims. Many alternatives, modifications, and variations will be apparent to those skilled in the art. Various changes may be made without departing from the spirit and scope of the invention, as defined in the following claims.

For example, although the present invention has been described in connection with the GATE_4 and GATE_5 embodiments, in which 4 dimensions and 5 dimensions of symbols are used, respectively, any number of dimensions (including only one dimension) may be used while still falling within the scope of the present invention. In general, as long as each token has more than one symbol, any number of symbols categorized in any number of dimensions may be used. Further, the GATE_4 and GATE_5 embodiments described above, as well as the associated screenshots, are meant to be illustrative and not to limit the scope of the present invention.

What is claimed is:

1. A method of allowing a user access to electronically stored information ("authenticating") using a predetermined electronically stored passcode ("passcode") that comprises a predetermined number of symbols ("passcode symbols") selected from a set of symbols, wherein each of the passcode symbols is characterized by a predetermined pin position, comprising:

presenting a token set to the user via a user interface of an electronic device, wherein the token set comprises at least two tokens, and wherein each token in the token set comprises at least two symbols that belong to the set of symbols;

requiring the user to select a token from the token set for each pin position in the passcode via the user interface; and

authenticating the user based on the tokens that the user selected, wherein the user is authenticated if:

the number of tokens selected by the user is equal to the number of symbols in the passcode,

at least one of the tokens selected contains a respective one of the passcode symbols, and

the pin position of each of the selected tokens that contains a respective one of the passcode symbols corresponds to the pin position of its respective passcode symbol in the passcode.

2. The method of claim 1, wherein the user is authenticated if:

the number of tokens selected by the user is equal to the number of symbols in the passcode;

each token selected contains a respective one of the passcode symbols; and

the pin position of each of the selected tokens corresponds to the pin position of each of the passcode symbols, based on which of the symbols in the passcode is included in each of the chosen tokens.